

BIT REDUCTION USING DITHER, ROUNDING AND ERROR FEEDBACK

This application claims the benefit of United Kingdom Application No. 0031771.9, filed December 29, 2000.

5

Field of the Invention

The present invention relates to data processing generally and, more particularly, data processing to reduce of the number of bits in a data word.

Background of the Invention

In data processing operations, particularly those involving some form of arithmetic calculation, it is usually desirable to truncate the number in the calculation in order to reduce the number of bits used to represent the result. This is necessary since registers in a computer only hold numbers of a fixed length bits. Arithmetical operations on such numbers can result in a number which has more bits than allowable. The number must then be reduced for further processing. The Least Significant Bits (LSBs) of the data word or number are typically removed. When dealing with graphics or video data, the bit reduction process can lead to unsightly quantisation-type visual effects when the data is displayed on screen.

FIG. 1 shows a diagrammatic representation of the bit reduction process where three 9-bit data streams R1, R2, R3 are processed in a processing block 10 to produce a 12-bit output data stream R. The data stream R is then is reduced in a bit reduction circuit 12 to an 8-bit data stream.

One way to reduce the bits of the 12-bit data word R to 8-bits is to remove the four LSBs from the 12-bit number. This is known as truncation.

FIGS. 2(a), 2(b) and 2(c) are graphs of the brightness of an object across a screen display where the object dims from left to right on the screen. One example would be an object provided with a shadow. The effect of truncation on the object edge is illustrated in Figure 2(a). The line 14 indicates the desired sharp edge and the line 16 represents the effect produced by truncation. This is, of course, greatly magnified in Figure 2(a). However, the effect could still be discernable by the eye.

Various techniques have already been developed and are widely used in bit reduction to reduce the impact of these visual effects and these are discussed below. For example, when rounding, half of the maximum number which can be represented by the LSBs is added to the number to be rounded and then the LSBs are discarded.

FIG. 3 illustrates an example of rounding of two 12-bit numbers 18, 20 to which the number 1000 is added to give two new 12-bit numbers 22, 24. When the four LSBs of the new numbers 22, 24 are discarded, the result, in the case of number 18, is a number which is the same as the number which would have been generated by discarding the four LSBs of number 18. However, with the number 20 the result is a different 8-bit number. The effect of rounding is shown at 26 in FIG. 2(b) where the ideal object edge 14 is formed by the 12-bit number.

A dither process is similar to rounding but, instead of the offset being half the maximum possible value of the four LSBs, the offset for each data word representing each pixel is varied from pixel to pixel.

FIG. 4 is a diagrammatic representation of an array 28 of pixels in a rectangular region of the screen display with the coordinates for the pixel in the top left hand corner of the

10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28

00-340
1496.00083

region being X, Y. The pixels in the rectangular block covered by coordinates X, Y to X +3, Y +1 represent the dimming edge of an object being drawn. In this example, the offset for the pixel at coordinate x, y would be one quarter of the maximum value of the four LSBs. Since the maximum value for the four LSBs is 1111, one quarter of this is 0100 and this offset value is 5 shown in the x, y pixel coordinate of FIG. 4. For the pixel for the coordinate x +1, y the offset would be one half (1000) of the maximum value for the LSBs. Moving along the horizontal row of pixels the offset would alternate between one quarter and one half, as can be seen from FIG. 4. Then, for the next row the offset for the pixel coordinate x, y+1 would be one half (1000) and that for x +1, y +1 would be three quarters (1011), again alternating pixel by pixel along the x axis. Thus, the offsets are indicated in binary form in the pixel squares of FIG. 4. The effect of dithering is to “fuzz” the changes in brightness (pixel) values to reduce the visible effects of quantisation, as shown at 30 in FIG. 2(c).

15 Error feedback can be used with any of the above processes (i.e., truncation, rounding or dither). Error feedback is diagrammatically represented in FIG. 1 by the feedback path 32. Using error feedback, some or all of the discarded LSBs from a 12-bit word are fed back and added to the next 12-bit word before the LSBs of the next 12-bit word are discarded. Error feedback has the effect of providing a smearing of brightness errors along the object edge.

Summary of the Invention

20 One aspect of the present invention provides a circuit for reducing the number of bits in a K bit value from K to N bits. The circuit generally comprises a first summing circuit, a control circuit, an error feedback circuit, a second summing circuit, and a processor. The first summing circuit may add an error offset value and the N + m MSB's of the K bit value to

00-340
1496.00083

produce a result data value. The control circuit may generate a dither offset value. The error feedback circuit may receive m LSBs of the result data value and generate an error value in dependence on the m LSBs. The second summing circuit may add the dither offset value and the error value to provide the error offset value. The processor may selectively control 5 generation of the dither offset value and the error value.

The present invention also provides a method of applying error concealment to the reduction of a data value, comprising the steps of (A) receiving a series of successive data values, (B) generating a respective error offset value for each of said data values and (C) adding each of said error offset values to a MSBs of the next following data value to produce a respective result value.

In one example, the K bit data value may be representative of a pixel data for display on a video display screen. The control circuit has a dither generating circuit for generating the dither offset value in dependence on n LSBs of each of the X and Y coordinates of the pixel data represented by the K bit data value.

15 The object, features and advantages of the present invention include a circuit which enables one or a combination of bit reduction techniques to be applied selectively to any video or graphics component.

Brief Description of the Drawings

20 These and other objects, features and advantages of the present invention will be apparent from the following detailed description and the appended claims and drawings in which:

FIG. 1 is a diagrammatic representation of a conventional bit reduction of a 12-bit number to an 8-bit number with error feedback;

00-340
1496.00083

FIG. 2(a), 2(b) and 2(c) are diagrammatic representations of the effects of various conventional techniques of bit reduction on the edge sharpness of an object drawn on a display screen;

5 FIG. 3 is an illustration of bit reduction on a number using rounding;

FIG. 4 is a representation of bit reduction using dither;

FIG. 5 is a schematic representation of bit reduction circuit according to a preferred embodiment of the present invention;

FIG. 6 is a schematic representation of one form of limit circuit for the circuit of

FIG. 5;

10 FIG. 7 is a table showing the relationship between the dither offset and the LSB of each pixel data number;

FIG. 8 is a representation of an array of pixels showing the dither offset for each pixel data number;

15 FIG. 9 is a schematic representation of one form of control circuit of the circuit of FIG. 5;

FIG. 10 is a schematic representation of a second form of control circuit for the circuit of FIG. 5;

FIG. 11 is a schematic representation of a third form of control circuit for the circuit of FIG. 5; and

20 FIG. 12 is a circuit in accordance with an alternate embodiment of the present invention.

Detailed Description of the Preferred Embodiments

Referring now to FIG. 5, a circuit 100 is shown in accordance with a preferred embodiment of the present invention. The circuit 100 may reduce a data word having K bits down to a data word having N bits. In one example, where K = 12, N = 8 and M = 2. However, the bit reduction described can be applied to other sizes of data words where at least a 2-bit reduction is required and M is 2 or more. The bit reduction of the circuit 100 may be applied to any video or graphics component such as the Red, Green, Blue, Luma, Alpha components. The circuit 100 enables a combination of truncation, rounding and dither, with or without error feedback, to be used for bit reduction of data words.

The circuit 100 has an input 102 for receiving each 12-bit word for bit reduction. Each data word has bits K-1:0 where, in this particular example, K is 12. The 12-bit word is truncated by removal of the two LSBs in a truncating process represented by 104 in FIG. 5. It will be appreciated that this simple truncation does not require any hardware to effect the truncation. The truncated data word comprises N+2 bits where, in this example, N is 8. The resulting truncated N+2 word is then fed to an input 106 of a summing circuit 108 which has a second input 110 connected to the output of a further summing circuit 112.

The summing circuit 112 has a first input 114 connected to a control block or circuit 116 which generates dither offset data bits. The output of the summing circuit 108 is applied both to a limit circuit 118 and to an error feedback register 120 whose output is in turn applied to a second input 122 of the summing circuit 112.

An error-concealing offset, which is a 3-bit word, is applied to the input 110 of the summing circuit 108 to be added to the N+2 bit word. The addition may overflow, depending on the expected range of the data signal being processed, with the result that the output of the

5
10
15
20

00-340
1496.00083

summing circuit 108 may be a $N+3$ bit word. This $N+3$ word is applied to the limit circuit 118 in a further truncated form where the two LSBs are discarded, leaving $N+1$ bit word which is received by the limit circuit 118. The error feedback register 120 receives the two LSBs of the $N+3$ bit word from the summing circuit 108 and these are fed back to input 122 of the summing circuit 112 as a 2-bit error signal. The output is then combined with the 2-bit dither signal from the control circuit 116 by the summing circuit 112 to provide the 3-bit value for the error-concealing offset. The error-concealing offset signal is applied to input 110 of the summing circuit 108 for addition to the next $N+2$ bit word from the truncation process 104.

The control circuit 116 generates a "clear" signal between horizontal lines scanned on the display screen and between frames. The clear signal is applied to the error feedback register 120 to clear the register. The clear signal ensures that the error bits are not carried across between non-adjacent values of $N+2$ bit words applied to the summing circuit 108. The clear signal may be a single bit with a value of 0 or 1. One value, (e.g., 1) clears the register 120 to prevent feedback while the other value (e.g., 0) allows feedback. A microprocessor (CPU) 40 applies a concealment type signal to the control circuit to set the form of error concealment which is to be applied by the circuit 100 to the data value being processed. Where error feedback is to be inhibited, the CPU 40 simply latches the clear signal to the value which clears the register 120. Effectively the error value 00 is applied to the summing circuit 112.

The operation of the circuit of FIG. 5 is now described. The first stage in processing a K bit word down to N bits (assuming that $K \geq (N+2)$) is to discard those LSBs which are so insignificant that they would not be useful for error concealment. Where a N bit output from the circuit of FIG. 5 is required and K is greater than $N+2$ bits then the K bit word is truncated to $N+2$ bits. This operation is represented at 104 in FIG. 5 but it will be appreciated

00-340
1496.00083

that this truncation does not require any hardware. If the input K bit word is only N+2 bits then the truncation process is not necessary and the K bit word is applied to the input 106 of the summing circuit 108.

An error-concealing offset is generated in a manner which is described later and is added to the N+2 bit word in the summing circuit 108. The error-concealing offset is here a 3-bit value and the addition in this example therefore may overflow, resulting in a N+3 bit word. Although the circuit 108 is described as a summing circuit, any suitable form of circuit may be used to provide the desired output value as a function of the two input values.

The two LSBs of the N+3 bit word are ignored and the N+1 MSBs of the result value are applied to the limit circuit 118. This detects illegal out of range values which were caused by the summing process and substitutes the maximum legal value when necessary.

FIG. 6 shows one example of circuit that the limit circuit 118 can take. Here, the limit circuit is a multiplexer 172 which has one input 174 to receive the MSB, one input 176 to receive the remaining N bits and an input 180 for receiving an N bit value of the form 11111111. Where bit N is 1 then the multiplexer outputs the N bit word as 11111111. Where bit N is 0 then the output word N is formed by the N LSBs of the input word N+1.

The two LSBs of the N+3 bit word are applied to the error feedback register 120. If the clear signal from the control circuit 116 is inactive, indicating that error feedback is required, then the two LSBs from the error feedback register 120 are fed to the summing circuit 112 and added to a 2-bit dither value which is also applied to the summing circuit 112. The result of the summation will be either a 2-bit or a 3-bit value although for this particular operation it is assumed that the result of the addition of the summing circuit 112 overflows to produce a 3-bit value. Although the circuit 112 is described as a summing circuit, any suitable

00-340
1496.00083

form of function generator may be used to provide the desired output value as a function of the two input values.

As will be appreciated, the 2-bit error value from the error feedback register 120 is formed by the stored two LSBs of the N+3 bit word from the summing circuit 108. The 2-bit error value is then added to the 2-bit dither value in the summing circuit 112 to form the error-concealing offset. The error-concealing offset is then added to the next following data value N+2 applied to the input 106 of the summing circuit 108.

Where the clear signal from the control circuit 116 is constantly active, error feedback from the error feedback register 120 is disabled. Ideally, the clear signal clears the register 120. If dither offset is applied to the summing circuit 108, in this example a 2-bit value may be used, although any suitable number of bits may be implemented to meet the design criteria of a particular implementation. In order for the control circuit 116 to produce the dither signal it needs to know the X, Y coordinate of each pixel or data value in the final image displayed on the screen, or at least the LSBs respectively of X and Y.

FIG. 7 is a table showing the dither offset which is generated using only the LSB X[0] and Y[0] of the X and Y coordinate values. As can be seen from FIG. 7, when the desired error concealment applied by the circuit 100 is truncation then the dither is fixed at 0. The 0 then ensures that no dither offset is generated by the dither source. If the desired error concealment applied by the circuit 100 is rounding then the dither offset is fixed at binary 10. The binary 10 is equivalent to half the maximum possible binary value of the two LSBs on the N bit output of the limit circuit 118. If the desired error concealment applied by the circuit 100 is dither then, as can be seen from the table of FIG. 7, where both the Y and X LSBs of the pixel coordinates are 0 then the dither offset is 01. For Y and X LSBs of 0 and 1 respectively then the dither offset

00-340
1496.00083

is 10. For Y and X coordinate LSBs of 1 and 0 respectively the dither offset is 10 and for Y and X coordinate LSBs of 1 and 1 then the dither offset is 11.

FIG. 8 is similar to FIG. 4 and is a diagrammatic representation of an array of 16 pixels in a rectangular region of the screen display. The dither offset for each pixel which is applied by the circuit of FIG. 5 is shown in the rectangle representing that pixel and as can be seen the binary value of the dither offset alternates between 01 and 10 along the first and third lines and between 10 and 11 along the second and fourth lines.

The regular pattern of dither offset shown in FIG. 8 has an average value of binary 10 added to the pixel display data. However, other arrangements are possible so long as the average value of the dither offset over the display area is equivalent to half the maximum value of the two LSBs of the N bit output word from the limit circuit 118.

As is mentioned above, in order for the control circuit 116 to produce the dither signal it needs to know the LSBs of the X, Y coordinate of each pixel or data value in the final image displayed on the screen, or at least the LSBs respectively of X and Y. The LSBs of the X, Y coordinate may be achieved in one of two ways.

FIG. 9 shows one example of the control circuit 116. The control circuit has a dither generation circuit 150 and a clear signal generation circuit 152. The dither generation circuit 150 is a set of logic gates with inputs 154, 156 each of which receives the LSB X[0] and Y[0] of the X and Y coordinate values respectively of the pixel data word K applied to the bit reduction circuit 100. The clear signal generation circuit 152 generates a "clear" signal when the X coordinate LSB is 0, or when the microprocessor (CPU) 40 applies a concealment type signal to the control circuit indicating error feedback is not required, and may be formed by a set of logic gates.

10
11
12
13
14
15
16
17
18
19
20

00-340
1496.00083

FIG. 10 shows a second form of the control circuit 116 in which the dither generation is calculated by reference to the horizontal (HSync) and vertical (VSync) synchronisation signals. In FIG. 10, separate X and Y counters 158, 160 are provided and the respective outputs of each are applied to the inputs 154, 156 of the dither generation circuit 150.

The HSync signal is applied to both counters 158, 160 whilst the VSync is applied to the Y counter 160. The X coordinate is incremented by a pixel clock 170, as a result of which the circuit 158 counts from an initial value, usually zero, for each successive pixel X coordinate. Each HSync pulse resets the circuit 158 between horizontal line scans when the X coordinate returns to its initial or base value and at the same time increments the Y counter 160 by one. The VSync signal resets the Y counter 160 between scanned fields of the image.

The X counter counts only through the X coordinates in response to the pixel clock pulses and applies the LSB of the X pixel coordinate to input 154 of the dither generation circuit 150. The Y coordinate is incremented at the end of each horizontal line by the HSync signal and the Y counter counts through the Y coordinates. The LSB of the HSync signal is applied to input 156 of the dither generation circuit 150. The latter then generates the dither offset value in accordance with the table of FIG. 7 and applies the dither signal to the input 114 of the summing circuit 112. It will be appreciated here that since both the X counter 158 and the Y counter 160 count through the pixel coordinates the dither generation circuit 150 can make use of more than one LSB from each, if desired.

FIG. 11 shows a circuit similar to FIG. 10 with the X and Y counters replaced by respective X and Y toggle bit circuits 162, 164. Each circuit is a single bit register which is toggled between 0 and 1, the bit representing the LSB of the X and Y pixel coordinates respectively. The X toggle bit circuit 162 is toggled between 0 and 1 by the pixel clock 170, as

00-340
1496.00083

a result of which the circuit 162 toggles from one state (0 or 1) to its other state (1 or 0) for each successive pixel X coordinate. Each HSync pulse resets the X toggle bit circuit 162 between horizontal line scans when the X coordinate returns to its initial or base value and at the same time toggles the Y toggle bit circuit 164 from one state (0 or 1) to its other state (1 or 0). The 5 VSync signal resets the Y toggle bit circuit 164 between scanned fields of the image. The circuit of FIG. 11 is useful where only one X and Y LSB is required for dither generation.

For the circuits of FIGS. 9, 10 and 11 the clear generation circuit 152 applies the clear signal to the error feedback register 120 either when the X coordinate LSB is 0 (FIG. 9) or a HSync pulse is received (FIGS. 9 and 10) or when the CPU applies the concealment type signal to the clear generation circuit 152 indicating that no error feedback is required, thus generating a permanent clear signal. In addition, the CPU 40 also applies the concealment type signal to the dither generation circuit 150. If dither offset is not to be used in the processing of a data word by the circuit 100 then the concealment type signal inhibits the generation of the dither offset signal from the dither generation circuit 150.

FIG. 12 shows a further bit reduction circuit 200 for use with multiplexed component data signals. The example circuit of FIG. 12 processes three data components in sequence although it will be appreciated that the number of components which are processed may be more or less than three. In this example, the circuit 200 processes three graphics components, for example Red, Green and Blue signals. The values for these components are applied to the 20 truncation process 104 in sequence. If the K bit values for these three components Red, Green and Blue are represented by K_R , K_G and K_B then the components are fed to the circuit 200 in the sequence:

$$K_R \ K_G \ K_B \ K_R \ K_G \ K_B \ K_R \ K_G \ K_B \dots$$

00-340
1496.00083

The circuit 200 is similar to the circuit 100 of FIG. 5 with the exception that the error feedback register 120 is replaced by three sets of error feedback registers 220R, 220G and 220B. Three sets of registers are necessary to ensure that the error value from one Red signal K_R is added through the summing circuits 112 and 108 to the next Red signal K_R and not to a Blue or a Green data signal. This also applies to the error value for the Green and Blue pixels.

5 Where the circuit is used to process multiplexed component data where there are two or more than three components then it will be appreciated that a respective register 220 is provided for each component.

While the invention has been particularly shown and described with reference to the preferred embodiments thereof, it will be understood by those skilled in the art that various changes in form and details may be made without departing from the spirit and scope of the invention.

50
49
48
47
46
45
44
43
42
41
40
39
38
37
36
35
34
33
32
31
30
29
28
27
26
25
24
23
22
21
20
19
18
17
16
15
14
13
12
11
10